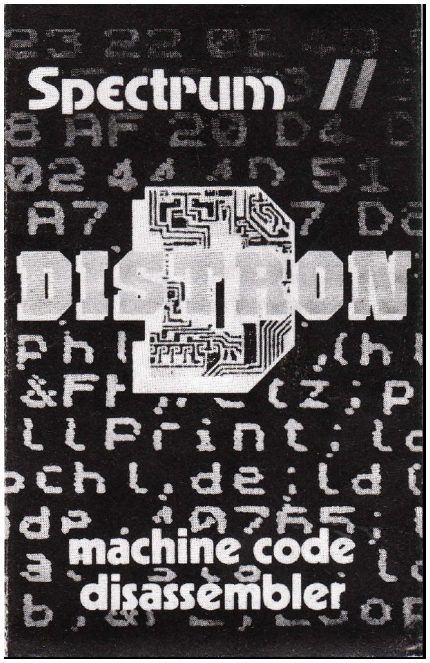

DISTRON

A full Z80 Dis-assembler written specially for the SPECTRUM. Standard Z80 mnemonics are used supplemented by mnemonics which are special to the SPECTRUM.

The following features are provided:

i Output can be in decimal or hexadecimal as you choose;
ii data bytes following calls to RST 08 and RST 28 are dis-assembled correctly;
iii output can be directed to the printer or the screen;
iv start address can be entered as decimal or hexadecimal;
v optional end address so that when output is sent to the printer, it stops automatically;
vi completely in machine-code, so needs no BASIC to run - hence can be loaded after loading other programs without needing to MERGE or re-load your program.
vii A re-locator program is supplied to enable DISTRON to be re-located anywhere in memory.

HOW TO LOAD DISTRON

Two versions of DISTRON are supplied, one for the 48k SPECTRUM and another for the 16k version. Choose the correct version for your SPECTRUM.

Two steps are needed: (16k version in brackets)

1. Type CLEAR 59999 (or CLEAR 29291 (16k))
2. Type Load " " CODE

and set your cassette running as normal.

HOW TO RUN DISTRON

To run the dis-assembler, type:
RANDOMIZE USR 60000 (RANDOMIZE USR 29292)
You will then obtain brief operating instructions. Before you can start dis-assembly, you will be asked to choose between decimal or hexadecimal output. Entering H or h will give output

in hexadecimal, and entering D or d will give output in decimal. Once you have chosen your preferred output, DISTRON will remember your choice, and if you run DISTRON again, it is only necessary to press ENTER and it will give you the same output as you chose last time.

You will then be asked to enter the Start Address. This is the address where dis-assembly is to start. Enter as a decimal number or as a hexadecimal number by prefixing the hex number with a &. For example, 9720 or &25f8 or &25F8 are all equivalent. This is independent of whether you selected decimal or hexadecimal output.

You will then be asked to enter the End address. This is only needed if you are sending the output to the printer, so that the printer stops at the end of the code you are dis-assembling - the paper is too expensive to waste! If you are sending the output to the screen, simply press ENTER.

Dis-assembly will now commence: the format of the output is as follows:

Address	*M/C Bytes*	*Mnemonic*	*for example:*
25F8	CD 30 25.....	call 2530	
25FB	28 28	jr z,2625	
25FD	ED 4B 76 5C	ld bc, (5C76)	
2601	CD 2B 2D	call 2D2B	
2604	EF	RST 28	
2605	A1	defb A1 = 1	
2606	OF	defb OF = ADD	
etc ...			

The above is an extract from the ROM inside the SPECTRUM, and is the section which deals with the RND function. It is interesting to compare the description of the RND function on page 74 of the SPECTRUM manual with the dis-assembled listing, as this illustrates the use of the data bytes which follow the RST 28 instruction. The RST 28 function on the SPECTRUM is the 'Floating point' calculator. Which type of calculation is determined by the data bytes which follow this instruction. The mnemonics which DISTRON uses are as follows:

data byte	*mnemonic*	*meaning*
00	JP-TR	jump-true
01	EXCH	exchange
02	DEL	delete
03	SUB	subtract
04	MPY	multiply
05	DIV	divide
06	POWER	raise to power
07	OR	or
08	&	and
09	LEQ	less than or equal
0A	GEQ	greater than or equal
0B	NEQ	not equal
0C	GTR	greater than
0D	LESS	less than
0E	EQL	equal
0F	ADD	add

10	$&	string and
11	$LEQ	string less than or equal
12	$GEQ	string greater than or equal
13	$NEQ	string not equal
14	$GTR	string greater
15	$LESS	string less
16	$EQL	string equal
17	$ADD	string add (concatenate)
18	VAL$	VAL$ function
19	USR$	USR$ function
1A	READ	read-in
1B	NEG	negate
1C	CODE	code
1D	VAL	VAL function
1E	LEN	LEN function
1F	SIN	SINe function
20	COS	COSine function
21	TAN	TANgent function
22	ASN	Arc-SiNe function
23	ACS	Arc-CoSine function
24	ATN	Arc-Tangent function
25	LN	Natural Logarithms - LN function
26	EXP	EXP function
27	INT	INTeger function
28	SQR	SQRoot function
29	SGN	SGN function
2A	ABS	ABS function
2B	PEEK	PEEK function
2C	IN	IN function
2D	USR	USR function
2E	STR$	STR$ function
2F	CHR$	CHR$ function
30	NOT	NOT function
31	DUP	duplicate
32	MOD	mod function - remainder after division
33	JMP	jump
34	STK	stack data - actual values follow
35	DJNZ	decrement and jump if non zero
36	<0	less than 0
37	>0	greater than zero
38	EXIT	end of FP calc.
39	ARGT	get argument
3A	TRUNC	truncate
3B	FPC-2	floating point calculator number 2
3C	E-FP	convert 'E' form to fp form
3D	RESTK	re-stack
86 etc	06 etc	series-06 etc
A0-A4	-	constants 0, 1, 0.5, PI/2, 10
C0-C5	ST-M0-5	store in memory 0 to 5
E0-E5	RCL-0 to RCL-5	recall from memory 0 to 5

For more information, please refer to Dr Ian Logan's book 'Understanding Your Spectrum' published by Melbourne House Publishers, 1982.

The other special instruction on the SPECTRUM is the RST 08, which is the error reporting subroutine. One data byte follows this call and is one less than the 'Report Code' (see Appendix B of the Sinclair manual). DISTRON correctly decodes these and displays the Report Code.

CONTROLS
To send the output to the printer, at any time during dis-assembly, press P. This will copy the screen and then continue printing until the End address is reached, or if you press S, the output will return to the screen.

To stop dis-assembly, simply press N or SPACE at the scroll? prompt.

To re-enter a new start address, press R.

RE-LOCATING DISTRON
DISTRON contains within itself a program which allows you to re-locate DISTRON at any address within memory. This may be necessary, for example, if you wish to use DISTRON with other utilities, such as ZXED (toolkit) or ASTRON (assembler).

Only DISTRON is re-located, not the re-locating program, so you will have to keep the original copy of DISTRON in addition to any copies made to run at other addresses.

The procedure is as follows: (16k version in brackets)
Type: RANDOMIZE USR 62812 (RANDOMIZE USR 32204)
Then, enter the new start address for DISTRON. DISTRON will then be re-located and can be run using this new address. Not that you can only re-locate to lower addresses otherwise error B will result. Also, RAMTOP, the system variable pointing to the top of the BASIC area will be lowered to make room for the code; if there is not enough room in memory (perhaps your BASIC program is too big) error M will result.

You can save this version (for your own use only please!) on tape as follows:
SAVE "DISTRON" CODE new address, 2912
(insert the new address in the command).

IMPORTANT
Please note that using DISTRON will clear all variables, so if you have stored some variables, save them on tape before using DISTRON.

Roy Eastwood
1983

Printed in England